Using gm.x to Check Memory Usage

You can use the gm.x tool to check the memory usage of jobs running on Pleiades, Aitken, Electra, or Endeavour. This tool reports the memory usage from each process at the end of a run. Optionally, gm.x can also report memory usage at a fixed interval during the run.

The gm.x tool is available in the /u/scicon/tools/bin directory.

TIP: Add the directory to your shell startup script so that you can invoke gm.x without typing the full path each time. For example:
`set path = ( $path /u/scicon/tools/bin )`
Use the **`-h`** option to find out what types of memory usage can be reported:

```
pfe21% gm.x -h
Usage: gm.x [-options] [-h|-he] a.out [args]

Options -
   -hwm     ; high water mark (VmHWM)
   -rss     ; resident memory size (VmRSS)
   -wrss    ; weighted memory size (WRSS)
   -cs:n  ; print memory usage every
```

Additional information about each option can be found with **`gm.x -he`**. Here are brief descriptions for some of the options:

-hwm (default)
: Reports the peak resident set size (high water mark) that the kernel captures for the process. Specifically, the status is taken from the value shown in the VmHWM field of the /proc//status file.

-wrss
: Calls the system function **`get_weighted_memory_size`**. (See **man get_weighted_memory_size** for more information about this function.)

-c
: Reports the memory usage during the run; useful in the event that a job fails to complete.

-memv
: Provided for backward compatibility; allows memory usage to be reported in a variable unit. Without this option, memory usage is reported in MB.

The gm.x tool can be used for either OpenMP or MPI applications (linked with HPE's MPT library). You do not need to recompile your application to use it.

To use gm.x for an MPI code, add **`gm.x`** just before the executable in the command line. For example:

```
mpiexec -np 4 gm.x ./a.out
Memory usage for (r1i1n0,pid=9767): 1.458 MB (rank=0)
Memory usage for (r1i1n0,pid=9768): 1.413 MB (rank=1)
Memory usage for (r1i1n0,pid=9770): 1.413 MB (rank=3)
Memory usage for (r1i1n0,pid=9769): 1.417 MB (rank=2)
```

You can feed the standard output into the gm_post.x script to compute the total memory used and the average memory used per process. For example:

```
mpiexec -np 4 gm.x ./a.out | gm_post.x -v
Memory usage for (r1i1n0,pid=9767): 1.458 MB (rank=0)
Memory usage for (r1i1n0,pid=9768): 1.413 MB (rank=1)
Memory usage for (r1i1n0,pid=9770): 1.413 MB (rank=3)
Memory usage for (r1i1n0,pid=9769): 1.417 MB (rank=2)

Number of nodes     = 1
```

```
Number of processes = 4
Processes per node  = 4
Total memory        = 5.701 MB

Memory per node     = 5.701 MB
Minimum node memory = 5.701 MB
Maximum node memory = 5.701 MB

Memory per process  = 1.425 MB
Minimum proc memory = 1.413 MB
Maximum proc memory = 1.458 MB
```

If you use **dplace** to pin the process, add **gm.x** after **dplace** in the command line:

```
mpiexec -np NN dplace -s1 gm.x ./a.out
```

The gm.x tool was developed by NAS staff member Henry Jin.

Article ID: 220
Last updated: 13 May, 2021
Revision: 18
Running Jobs with PBS -> Optimizing/Troubleshooting -> Managing Memory -> Using gm.x to Check Memory Usage
https://www.nas.nasa.gov/hecc/support/kb/entry/220/